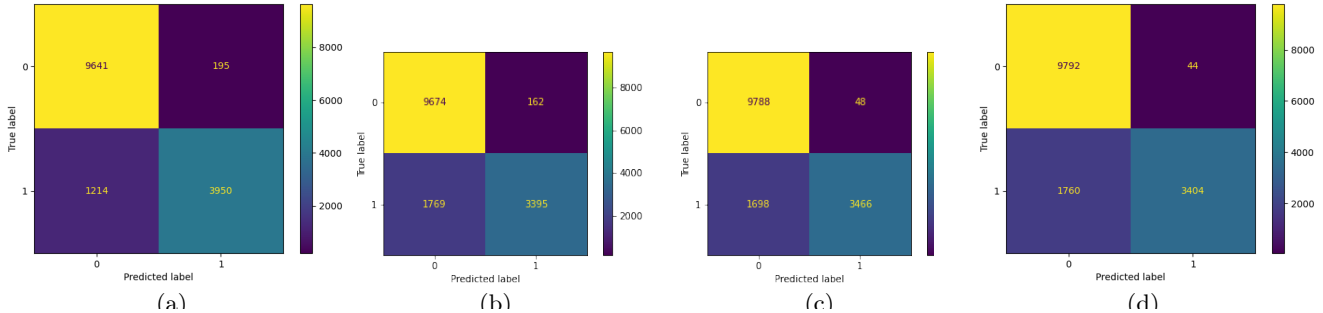
Model on FakeNewsCorpus without metadata

	accuracy	precision macro	recall macro	f1-score macro	support macro	precision weighted	recall weighted	f1-score weighted	support weighted
_TFIDF_LogisticRegression(solver='sag')_	0.906067	0.920559	0.872543	0.890272	15000.0	0.910468	0.906067	0.903239	15000.0
_ngram_2_LogisticRegression(solver='sag')_	0.871267	0.899932	0.820483	0.843917	15000.0	0.882949	0.871267	0.864267	15000.0
_ngram_3_LogisticRegression(max_iter=10000, n_j...	0.883600	0.919254	0.833153	0.858457	15000.0	0.898359	0.883600	0.877038	15000.0
_ngram_2_SVC(kernel='linear')_	0.879733	0.917442	0.827353	0.853089	15000.0	0.895703	0.879733	0.872576	15000.0
_ngram_3_SVC(kernel='linear')_	0.883933	0.920971	0.832993	0.858640	15000.0	0.899390	0.883933	0.877264	15000.0
_ngram_2_DecisionTreeClassifier()_	0.837667	0.822000	0.815145	0.818337	15000.0	0.836313	0.837667	0.836794	15000.0
_ngram_3_DecisionTreeClassifier()_	0.878867	0.905265	0.831337	0.854169	15000.0	0.889194	0.878867	0.872861	15000.0
_ngramKNN_2k=1	0.776267	0.753521	0.764694	0.757915	15000.0	0.783139	0.776267	0.778675	15000.0
_ngramKNN_2k=3	0.838333	0.834065	0.799326	0.812010	15000.0	0.837108	0.838333	0.833920	15000.0
_ngramKNN_2k=5	0.850533	0.860903	0.803662	0.821928	15000.0	0.854313	0.850533	0.844157	15000.0
_ngramKNN_2k=7	0.857467	0.877749	0.806833	0.828007	15000.0	0.865512	0.857467	0.850178	15000.0
_ngramKNN_2k=10	0.861933	0.897945	0.805180	0.830074	15000.0	0.877718	0.861933	0.852991	15000.0
_ngramKNN_3k=1	0.862000	0.871524	0.819258	0.836884	15000.0	0.865257	0.862000	0.856820	15000.0
_ngramKNN_3k=3	0.878600	0.909864	0.828696	0.852848	15000.0	0.891323	0.878600	0.872021	15000.0
_ngramKNN_3k=5	0.879600	0.911909	0.829505	0.853932	15000.0	0.892811	0.879600	0.873003	15000.0
_ngramKNN_3k=7	0.881400	0.917750	0.830049	0.855495	15000.0	0.896601	0.881400	0.874552	15000.0
_ngramKNN_3k=10	0.880667	0.918760	0.828386	0.854229	15000.0	0.896799	0.880667	0.873565	15000.0
_TFIDF_LogisticRegression(solver='sag')_	0.906067	0.920559	0.872543	0.890272	15000.0	0.910468	0.906067	0.903239	15000.0
_TFIDF_SVC(kernel='linear')_	0.916267	0.926137	0.888461	0.903315	15000.0	0.918827	0.916267	0.914337	15000.0
_TFIDF_DecisionTreeClassifier()_	0.873600	0.861547	0.856893	0.859126	15000.0	0.872937	0.873600	0.873191	15000.0
_KNN_1	0.691267	0.826811	0.552297	0.500293	15000.0	0.781167	0.691267	0.596511	15000.0
_KNN_3	0.680067	0.833357	0.535433	0.468258	15000.0	0.783142	0.680067	0.572787	15000.0
_KNN_5	0.677800	0.835265	0.532049	0.461624	15000.0	0.783955	0.677800	0.567881	15000.0
_KNN_7	0.676267	0.834740	0.529822	0.457298	15000.0	0.783267	0.676267	0.564668	15000.0
_KNN_10	0.674600	0.834171	0.527401	0.452558	15000.0	0.782521	0.674600	0.561150	15000.0

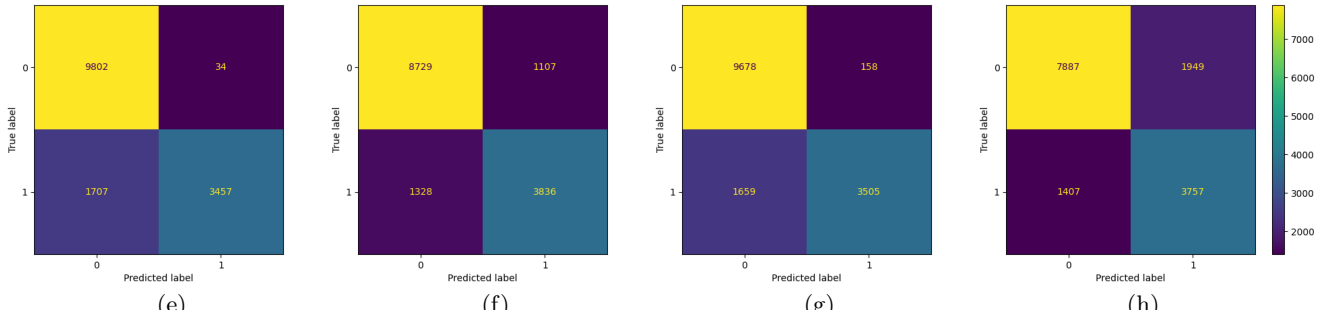


(a)

(b)

(c)

(d)

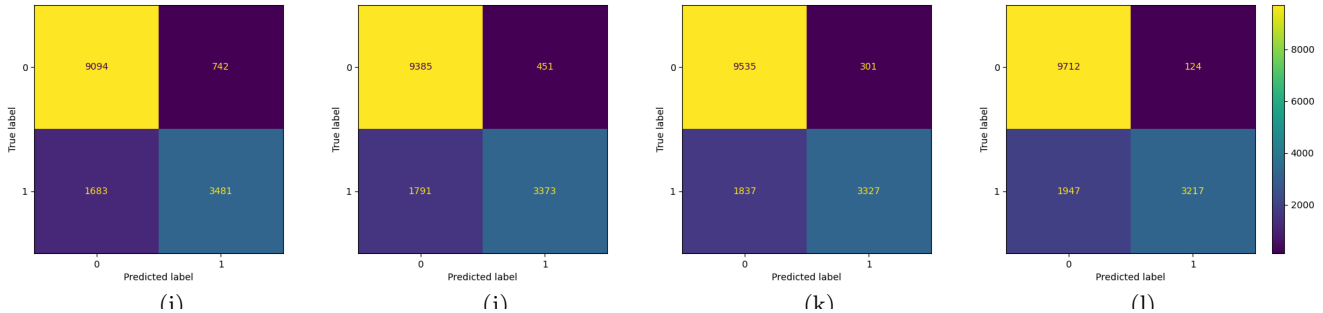


(e)

(f)

(g)

(h)

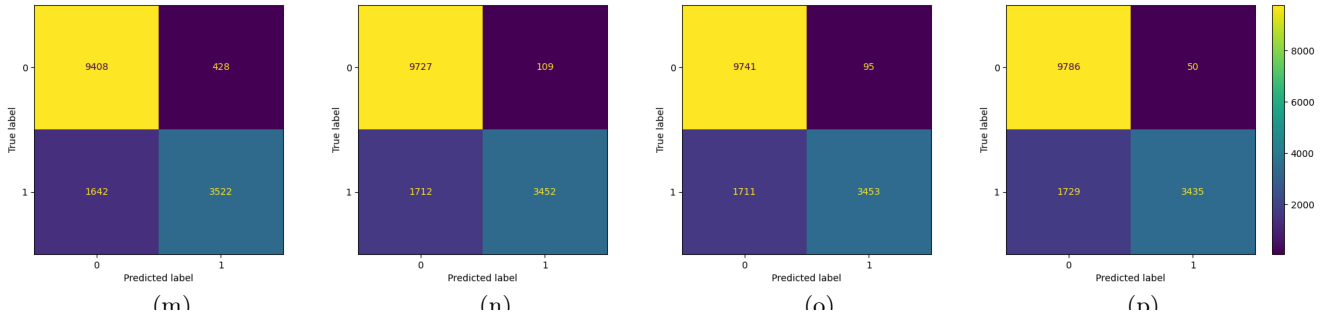


(i)

(j)

(k)

(l)

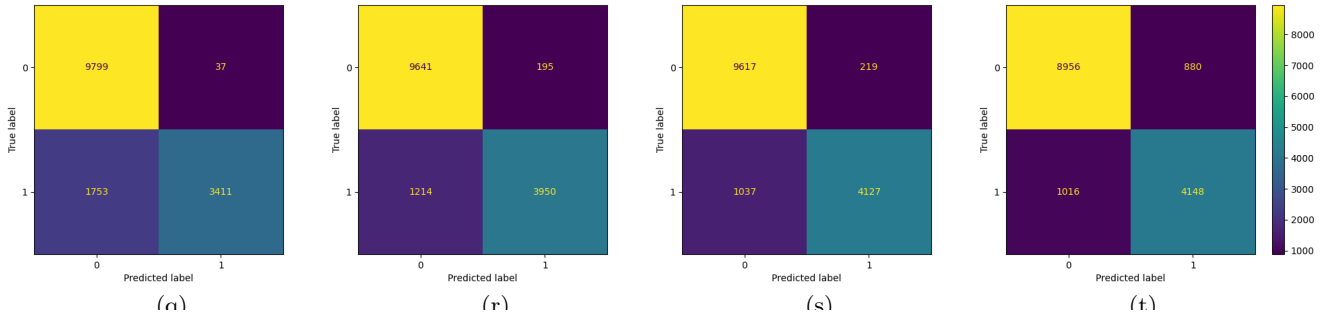


(m)

(n)

(o)

(p)

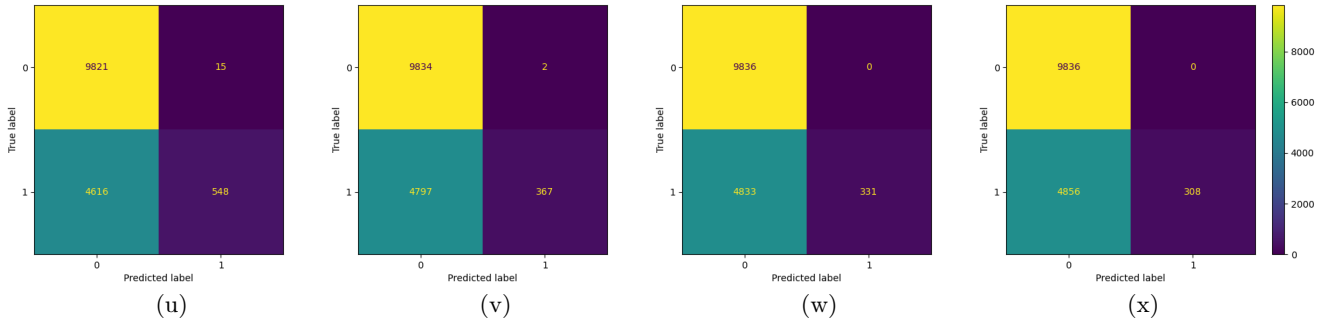


(q)

(r)

(s)

(t)



(u)

(v)

(w)

(x)

Figure 1: The models used in the above confusion matrices are respectively (a) bigram vectoriser with logistic regression; (b)

Model on FakeNewsCorpus with metadata

	accuracy	precision macro	recall macro	f1-score macro	support macro	precision weighted	recall weighted	f1-score weighted	support weighted
_TFIDF_LogisticRegression(solver='sag')_	0.935133	0.945112	0.911999	0.925577	15000.0	0.937442	0.935133	0.933883	15000.0
_ngram_2_LogisticRegression(solver='sag')_	0.873267	0.901814	0.823112	0.846528	15000.0	0.884824	0.873267	0.866481	15000.0
_ngram_3_LogisticRegression(max_iter=10000, n_j...	0.888733	0.922380	0.840562	0.865385	15000.0	0.902283	0.888733	0.882847	15000.0
_ngram_2_SVC(kernel='linear')_	0.893867	0.924631	0.848385	0.872387	15000.0	0.905808	0.893867	0.888694	15000.0
_ngram_3_SVC(kernel='linear')_	0.890400	0.923989	0.842707	0.867517	15000.0	0.903860	0.890400	0.884666	15000.0
_ngram_2_DecisionTreeClassifier()_	0.855200	0.841851	0.834539	0.837950	15000.0	0.854041	0.855200	0.854418	15000.0
_ngram_3_DecisionTreeClassifier()_	0.886867	0.910872	0.842772	0.864791	15000.0	0.895818	0.886867	0.881808	15000.0
_ngramKNN_2k=1	0.787200	0.764841	0.775100	0.769060	15000.0	0.792802	0.787200	0.789219	15000.0
_ngramKNN_2k=3	0.839533	0.833593	0.802725	0.814367	15000.0	0.837947	0.839533	0.835656	15000.0
_ngramKNN_2k=5	0.855200	0.864475	0.810532	0.828291	15000.0	0.858460	0.855200	0.849463	15000.0
_ngramKNN_2k=7	0.860133	0.877067	0.812178	0.832376	15000.0	0.866569	0.860133	0.853621	15000.0
_ngramKNN_2k=10	0.866933	0.903214	0.811522	0.836633	15000.0	0.882663	0.866933	0.858547	15000.0
_ngramKNN_3k=1	0.866733	0.875959	0.825627	0.842952	15000.0	0.869803	0.866733	0.861987	15000.0
_ngramKNN_3k=3	0.883400	0.911983	0.836081	0.859515	15000.0	0.894636	0.883400	0.877557	15000.0
_ngramKNN_3k=5	0.884733	0.916377	0.836362	0.860639	15000.0	0.897428	0.884733	0.878688	15000.0
_ngramKNN_3k=7	0.885267	0.918237	0.836447	0.861079	15000.0	0.898604	0.885267	0.879134	15000.0
_ngramKNN_3k=10	0.885400	0.921708	0.835169	0.860655	15000.0	0.900424	0.885400	0.878945	15000.0
_TFIDF_LogisticRegression(solver='sag')_	0.935133	0.945112	0.911999	0.925577	15000.0	0.937442	0.935133	0.933883	15000.0
_TFIDF_SVC(kernel='linear')_	0.940133	0.945249	0.921882	0.932006	15000.0	0.941056	0.940133	0.939328	15000.0
_TFIDF_DecisionTreeClassifier()_	0.920933	0.913663	0.910645	0.912121	15000.0	0.920698	0.920933	0.920789	15000.0
_KNN_1	0.783600	0.761456	0.751797	0.756011	15000.0	0.780558	0.783600	0.781565	15000.0
_KNN_3	0.798800	0.791599	0.747612	0.761111	15000.0	0.796243	0.798800	0.790665	15000.0
_KNN_5	0.809333	0.817952	0.749435	0.767125	15000.0	0.812981	0.809333	0.798005	15000.0
_KNN_7	0.813733	0.837325	0.747088	0.767309	15000.0	0.824697	0.813733	0.799681	15000.0
_KNN_10	0.809267	0.865305	0.728827	0.750518	15000.0	0.838822	0.809267	0.788225	15000.0

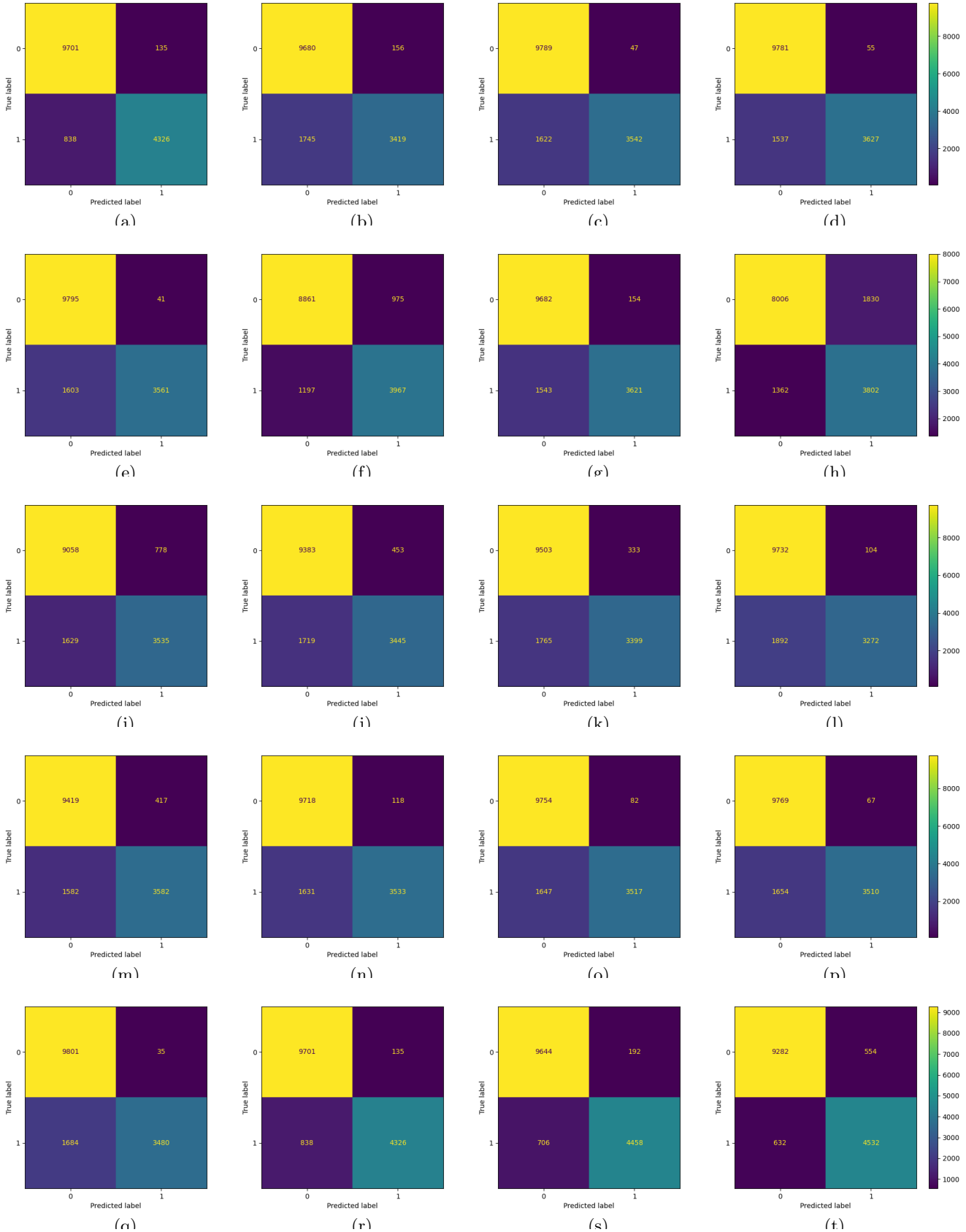


Figure 2: The models used in the above confusion matrices are respectively (a) bigram vectoriser with logistic regression; (b)

Model on FakeNewsCorpus only with metadata

	accuracy	precision macro	recall macro	f1-score macro	support macro	precision weighted	recall weighted	f1-score weighted	support weighted
_TFIDF_LogisticRegression(solver='sag')_	0.794400	0.804031	0.728069	0.745114	15000.0	0.798750	0.794400	0.780024	15000.0
_ngram_2_LogisticRegression(solver='sag')_	0.723667	0.811895	0.603309	0.587649	15000.0	0.779071	0.723667	0.661413	15000.0
_ngram_3_LogisticRegression(max_iter=10000, n_j...	0.666333	0.803556	0.515855	0.430138	15000.0	0.759771	0.666333	0.544408	15000.0
_ngram_2_SVC(kernel='linear')_	0.723533	0.816470	0.602471	0.586029	15000.0	0.782059	0.723533	0.660340	15000.0
_ngram_3_SVC(kernel='linear')_	0.666067	0.788685	0.515744	0.430350	15000.0	0.749517	0.666067	0.544483	15000.0
_ngram_2_DecisionTreeClassifier()_	0.722733	0.800465	0.603517	0.588786	15000.0	0.771295	0.722733	0.661885	15000.0
_ngram_3_DecisionTreeClassifier()_	0.665867	0.781431	0.515591	0.430259	15000.0	0.744501	0.665867	0.544375	15000.0
_ngramKNN_2k=1	0.502400	0.619851	0.594823	0.496140	15000.0	0.688306	0.502400	0.478647	15000.0
_ngramKNN_2k=3	0.509267	0.622609	0.599461	0.504018	15000.0	0.690866	0.509267	0.488127	15000.0
_ngramKNN_2k=5	0.718000	0.762228	0.602805	0.590760	15000.0	0.745088	0.718000	0.661834	15000.0
_ngramKNN_2k=7	0.718600	0.772631	0.601607	0.588013	15000.0	0.751970	0.718600	0.660257	15000.0
_ngramKNN_2k=10	0.721000	0.804561	0.600126	0.583193	15000.0	0.773523	0.721000	0.657841	15000.0
_ngramKNN_3k=1	0.390333	0.645575	0.532597	0.333986	15000.0	0.734580	0.390333	0.273649	15000.0
_ngramKNN_3k=3	0.390800	0.643272	0.532722	0.334974	15000.0	0.731535	0.390800	0.274961	15000.0
_ngramKNN_3k=5	0.390533	0.649070	0.533025	0.333951	15000.0	0.739103	0.390533	0.273485	15000.0
_ngramKNN_3k=7	0.389067	0.652017	0.532229	0.331321	15000.0	0.743099	0.389067	0.270117	15000.0
_ngramKNN_3k=10	0.388600	0.646948	0.531505	0.331068	15000.0	0.736555	0.388600	0.269966	15000.0
_TFIDF_LogisticRegression(solver='sag')_	0.794400	0.804031	0.728069	0.745114	15000.0	0.798750	0.794400	0.780024	15000.0
_TFIDF_SVC(kernel='linear')_	0.795133	0.811350	0.725317	0.743095	15000.0	0.802770	0.795133	0.779108	15000.0
_TFIDF_DecisionTreeClassifier()_	0.754400	0.728096	0.716839	0.721465	15000.0	0.749870	0.754400	0.751297	15000.0
_KNN_1	0.730267	0.712028	0.653550	0.661827	15000.0	0.721685	0.730267	0.709212	15000.0
_KNN_3	0.738200	0.765252	0.638352	0.641480	15000.0	0.753667	0.738200	0.699480	15000.0
_KNN_5	0.672600	0.632232	0.624710	0.627389	15000.0	0.664873	0.672600	0.667815	15000.0
_KNN_7	0.676933	0.635025	0.622173	0.625819	15000.0	0.665542	0.676933	0.668894	15000.0
_KNN_10	0.701267	0.670401	0.615387	0.617762	15000.0	0.685824	0.701267	0.673408	15000.0

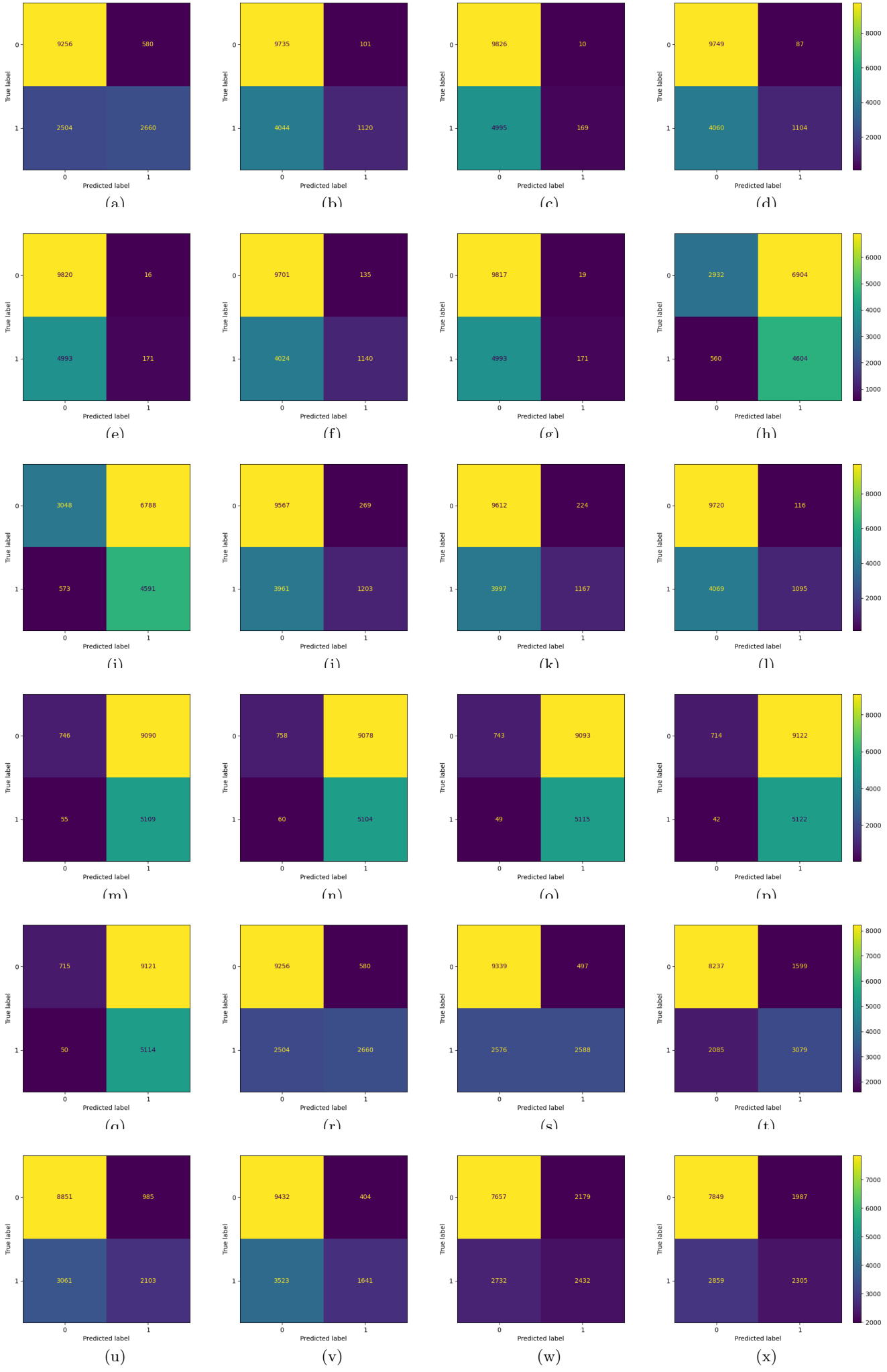


Figure 3: The models used in the above confusion matrices are respectively (a) bigram vectoriser with logistic regression; (b)

Model on LIAR dataset without metadata

	accuracy	precision macro	recall macro	f1-score macro	support macro	precision weighted	recall weighted	f1-score weighted	support weighted
_TFIDF_LogisticRegression(solver='sag')_	0.719390	0.511181	0.500389	0.425609	12790.0	0.605032	0.719390	0.608166	12790.0
_ngram_2_LogisticRegression(solver='sag')_	0.719703	0.457082	0.499134	0.421204	12790.0	0.574750	0.719703	0.605926	12790.0
_ngram_3_LogisticRegression(max_iter=10000, n_j...	0.722205	0.361102	0.500000	0.419349	12790.0	0.521580	0.722205	0.605712	12790.0
_ngram_2_SVC(kernel='linear')_	0.722205	0.361102	0.500000	0.419349	12790.0	0.521580	0.722205	0.605712	12790.0
_ngram_3_SVC(kernel='linear')_	0.722127	0.527776	0.500032	0.419597	12790.0	0.614188	0.722127	0.605819	12790.0
_ngram_2_DecisionTreeClassifier()_	0.700704	0.493692	0.498623	0.448409	12790.0	0.594988	0.700704	0.614194	12790.0
_ngram_3_DecisionTreeClassifier()_	0.721814	0.472194	0.499903	0.419764	12790.0	0.583284	0.721814	0.605812	12790.0
_ngramKNN_2k=1	0.466380	0.504129	0.504998	0.456092	12790.0	0.603190	0.466380	0.489336	12790.0
_ngramKNN_2k=3	0.700313	0.503242	0.500777	0.454155	12790.0	0.600697	0.700313	0.617056	12790.0
_ngramKNN_2k=5	0.710477	0.485769	0.498375	0.434249	12790.0	0.590546	0.710477	0.609932	12790.0
_ngramKNN_2k=7	0.717748	0.495203	0.499772	0.426572	12790.0	0.596044	0.717748	0.608166	12790.0
_ngramKNN_2k=10	0.721736	0.503964	0.500022	0.420283	12790.0	0.600956	0.721736	0.606064	12790.0
_ngramKNN_3k=1	0.719077	0.486015	0.499567	0.423649	12790.0	0.590902	0.719077	0.607029	12790.0
_ngramKNN_3k=3	0.721423	0.454796	0.499718	0.419904	12790.0	0.573585	0.721423	0.605766	12790.0
_ngramKNN_3k=5	0.720954	0.410963	0.499307	0.419472	12790.0	0.549158	0.720954	0.605392	12790.0
_ngramKNN_3k=7	0.721736	0.361037	0.499675	0.419191	12790.0	0.521486	0.721736	0.605483	12790.0
_ngramKNN_3k=10	0.721970	0.361070	0.499838	0.419270	12790.0	0.521533	0.721970	0.605597	12790.0
_TFIDF_LogisticRegression(solver='sag')_	0.719390	0.511181	0.500389	0.425609	12790.0	0.605032	0.719390	0.608166	12790.0
_TFIDF_SVC(kernel='linear')_	0.718452	0.518988	0.500952	0.428906	12790.0	0.609471	0.718452	0.609622	12790.0
_TFIDF_DecisionTreeClassifier()_	0.655825	0.495253	0.497082	0.483058	12790.0	0.595505	0.655825	0.615869	12790.0
_KNN_1	0.721032	0.497460	0.499967	0.421397	12790.0	0.597333	0.721032	0.606439	12790.0
_KNN_3	0.722127	0.361092	0.499946	0.419323	12790.0	0.521564	0.722127	0.605674	12790.0
_KNN_5	0.722205	0.361102	0.500000	0.419349	12790.0	0.521580	0.722205	0.605712	12790.0
_KNN_7	0.722205	0.361102	0.500000	0.419349	12790.0	0.521580	0.722205	0.605712	12790.0
_KNN_10	0.722205	0.361102	0.500000	0.419349	12790.0	0.521580	0.722205	0.605712	12790.0

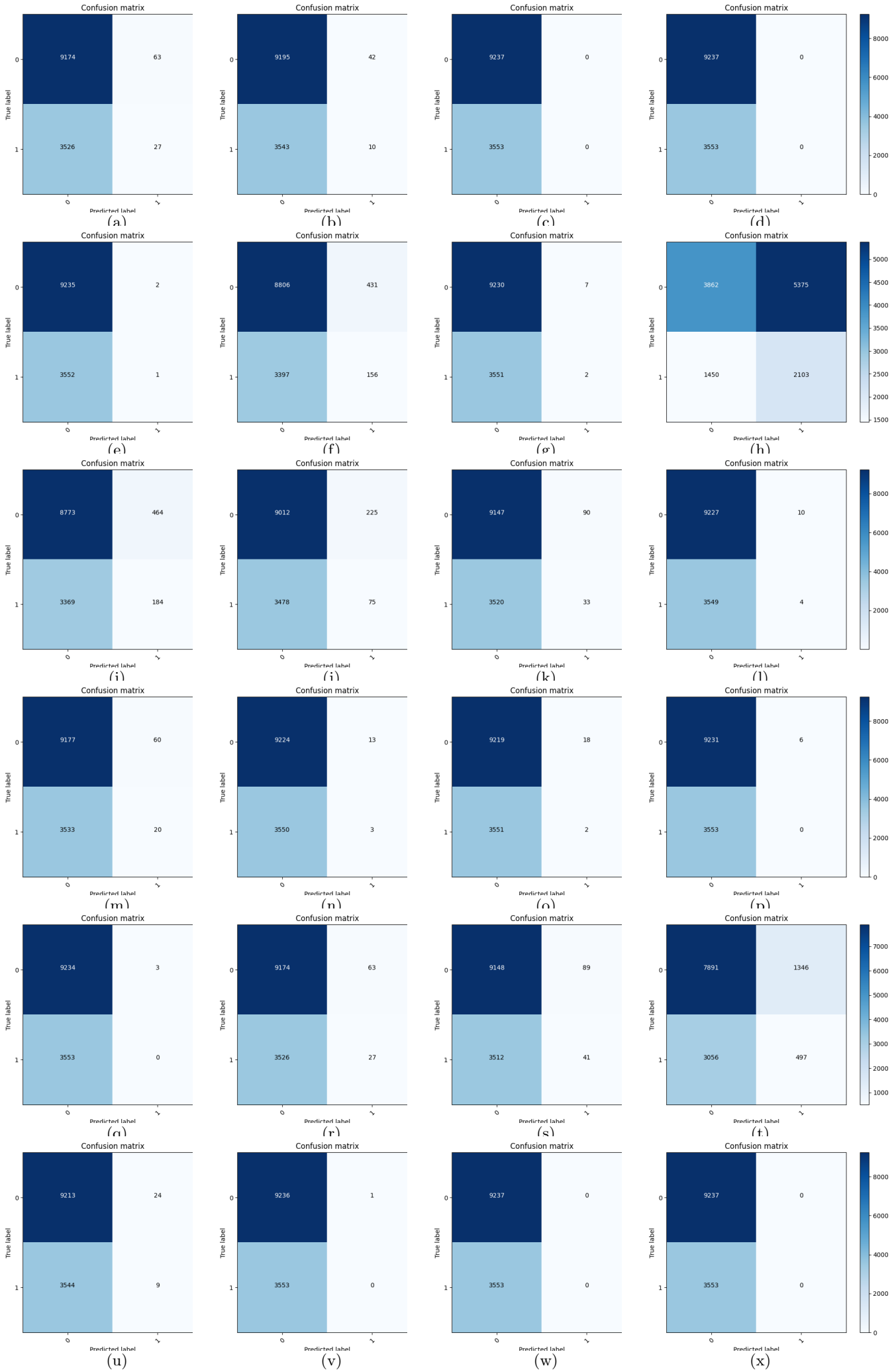


Figure 4: The models used in the above confusion matrices are respectively (a) bigram vectoriser with logistic regression; (b)

Model on LIAR dataset with metadata

	accuracy	precision macro	recall macro	f1-score macro	support macro	precision weighted	recall weighted	f1-score weighted	support weighted
_TFIDF_LogisticRegression(solver='sag')_	0.720094	0.453971	0.499231	0.420802	12790.0	0.573039	0.720094	0.605833	12790.0
_ngram_2_LogisticRegression(solver='sag')_	0.719781	0.438664	0.498928	0.420424	12790.0	0.564481	0.719781	0.605536	12790.0
_ngram_3_LogisticRegression(max_iter=10000, n_j...	0.722205	0.361102	0.500000	0.419349	12790.0	0.521580	0.722205	0.605712	12790.0
_ngram_2_SVC(kernel='linear')_	0.722205	0.361102	0.500000	0.419349	12790.0	0.521580	0.722205	0.605712	12790.0
_ngram_3_SVC(kernel='linear')_	0.722127	0.527776	0.500032	0.419597	12790.0	0.614188	0.722127	0.605819	12790.0
_ngram_2_DecisionTreeClassifier()_	0.695387	0.495426	0.498753	0.455031	12790.0	0.595973	0.695387	0.615872	12790.0
_ngram_3_DecisionTreeClassifier()_	0.721892	0.511111	0.500043	0.420063	12790.0	0.604931	0.721892	0.605995	12790.0
_ngramKNN_2k=1	0.478264	0.497320	0.496686	0.460938	12790.0	0.595968	0.478264	0.503887	12790.0
_ngramKNN_2k=3	0.672791	0.505078	0.502593	0.482560	12790.0	0.602094	0.672791	0.621989	12790.0
_ngramKNN_2k=5	0.706802	0.486632	0.497996	0.438631	12790.0	0.590954	0.706802	0.611062	12790.0
_ngramKNN_2k=7	0.709930	0.486158	0.498343	0.434987	12790.0	0.590757	0.709930	0.610147	12790.0
_ngramKNN_2k=10	0.719625	0.475741	0.499426	0.422248	12790.0	0.585169	0.719625	0.606456	12790.0
_ngramKNN_3k=1	0.719547	0.496216	0.499891	0.423817	12790.0	0.596628	0.719547	0.607264	12790.0
_ngramKNN_3k=3	0.720954	0.444331	0.499480	0.420016	12790.0	0.567728	0.720954	0.605681	12790.0
_ngramKNN_3k=5	0.722127	0.361092	0.499946	0.419323	12790.0	0.521564	0.722127	0.605674	12790.0
_ngramKNN_3k=7	0.722127	0.361092	0.499946	0.419323	12790.0	0.521564	0.722127	0.605674	12790.0
_ngramKNN_3k=10	0.722205	0.361102	0.500000	0.419349	12790.0	0.521580	0.722205	0.605712	12790.0
_TFIDF_LogisticRegression(solver='sag')_	0.720094	0.453971	0.499231	0.420802	12790.0	0.573039	0.720094	0.605833	12790.0
_TFIDF_SVC(kernel='linear')_	0.718452	0.491032	0.499653	0.425004	12790.0	0.593705	0.718452	0.607554	12790.0
_TFIDF_DecisionTreeClassifier()_	0.677717	0.504719	0.502193	0.478425	12790.0	0.601808	0.677717	0.621706	12790.0
_KNN_1	0.600000	0.496103	0.496189	0.496088	12790.0	0.595656	0.600000	0.597781	12790.0
_KNN_3	0.646599	0.485033	0.490348	0.476955	12790.0	0.588403	0.646599	0.609335	12790.0
_KNN_5	0.657780	0.486077	0.492113	0.473632	12790.0	0.589396	0.657780	0.611992	12790.0
_KNN_7	0.669038	0.485457	0.493067	0.467620	12790.0	0.589286	0.669038	0.613147	12790.0
_KNN_10	0.691556	0.490703	0.497226	0.455727	12790.0	0.593056	0.691556	0.614945	12790.0

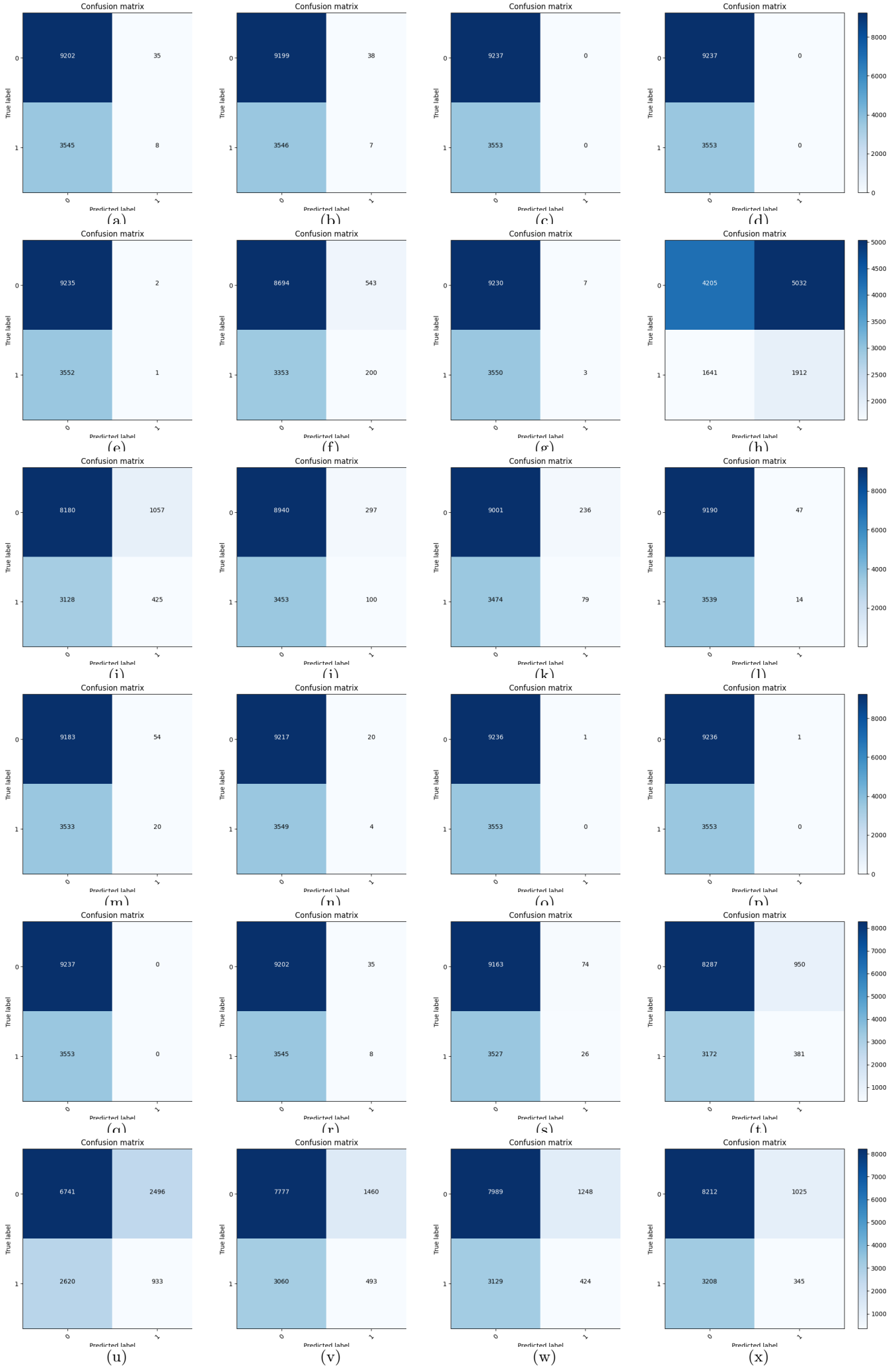


Figure 5: The models used in the above confusion matrices are respectively (a) bigram vectoriser with logistic regression; (b)

Model on LIAR dataset only with metadata

	accuracy	precision macro	recall macro	f1-score macro	support macro	precision weighted	recall weighted	f1-score weighted	support weighted
_TFIDF_LogisticRegression(solver='sag')_	0.679593	0.474018	0.490589	0.453367	12790.0	0.582493	0.679593	0.609647	12790.0
_ngram_2_LogisticRegression(solver='sag')_	0.702033	0.493106	0.498591	0.446643	12790.0	0.594657	0.702033	0.613709	12790.0
_ngram_3_LogisticRegression(max_iter=10000, n_j...	0.720954	0.582112	0.504243	0.434492	12790.0	0.645135	0.720954	0.613362	12790.0
_ngram_2_SVC(kernel='linear')_	0.713995	0.534108	0.504361	0.445495	12790.0	0.618499	0.713995	0.616973	12790.0
_ngram_3_SVC(kernel='linear')_	0.720407	0.572267	0.504037	0.434784	12790.0	0.639629	0.720407	0.613345	12790.0
_ngram_2_DecisionTreeClassifier()_	0.711493	0.527402	0.504188	0.448526	12790.0	0.614746	0.711493	0.617763	12790.0
_ngram_3_DecisionTreeClassifier()_	0.719937	0.566330	0.504059	0.435610	12790.0	0.636335	0.719937	0.613636	12790.0
_ngramKNN_2k=1	0.432760	0.491684	0.490381	0.428305	12790.0	0.589444	0.432760	0.450732	12790.0
_ngramKNN_2k=3	0.438311	0.495922	0.495264	0.433522	12790.0	0.594200	0.438311	0.456670	12790.0
_ngramKNN_2k=5	0.703675	0.534445	0.509513	0.470841	12790.0	0.619690	0.703675	0.626832	12790.0
_ngramKNN_2k=7	0.707584	0.530899	0.506591	0.459613	12790.0	0.617147	0.707584	0.622294	12790.0
_ngramKNN_2k=10	0.714855	0.504359	0.500367	0.433078	12790.0	0.601239	0.714855	0.610700	12790.0
_ngramKNN_3k=1	0.293901	0.485851	0.497209	0.252063	12790.0	0.578833	0.293901	0.173449	12790.0
_ngramKNN_3k=3	0.294371	0.481381	0.496061	0.253722	12790.0	0.572591	0.294371	0.176319	12790.0
_ngramKNN_3k=5	0.294527	0.482653	0.496343	0.253830	12790.0	0.574376	0.294527	0.176386	12790.0
_ngramKNN_3k=7	0.293354	0.484499	0.497003	0.251160	12790.0	0.576917	0.293354	0.172164	12790.0
_ngramKNN_3k=10	0.294840	0.485588	0.496992	0.253987	12790.0	0.578494	0.294840	0.176404	12790.0
_TFIDF_LogisticRegression(solver='sag')_	0.679593	0.474018	0.490589	0.453367	12790.0	0.582493	0.679593	0.609647	12790.0
_TFIDF_SVC(kernel='linear')_	0.689601	0.475076	0.492928	0.447956	12790.0	0.583560	0.689601	0.610271	12790.0
_TFIDF_DecisionTreeClassifier()_	0.607115	0.497979	0.498084	0.497717	12790.0	0.597168	0.607115	0.601892	12790.0
_KNN_1	0.700547	0.512728	0.503278	0.459836	12790.0	0.606440	0.700547	0.620085	12790.0
_KNN_3	0.718452	0.536560	0.502251	0.432743	12790.0	0.619469	0.718452	0.611654	12790.0
_KNN_5	0.602502	0.496959	0.497055	0.496889	12790.0	0.596347	0.602502	0.599330	12790.0
_KNN_7	0.616263	0.496696	0.497056	0.495580	12790.0	0.596230	0.616263	0.605228	12790.0
_KNN_10	0.668804	0.501475	0.500785	0.481898	12790.0	0.599729	0.668804	0.620192	12790.0

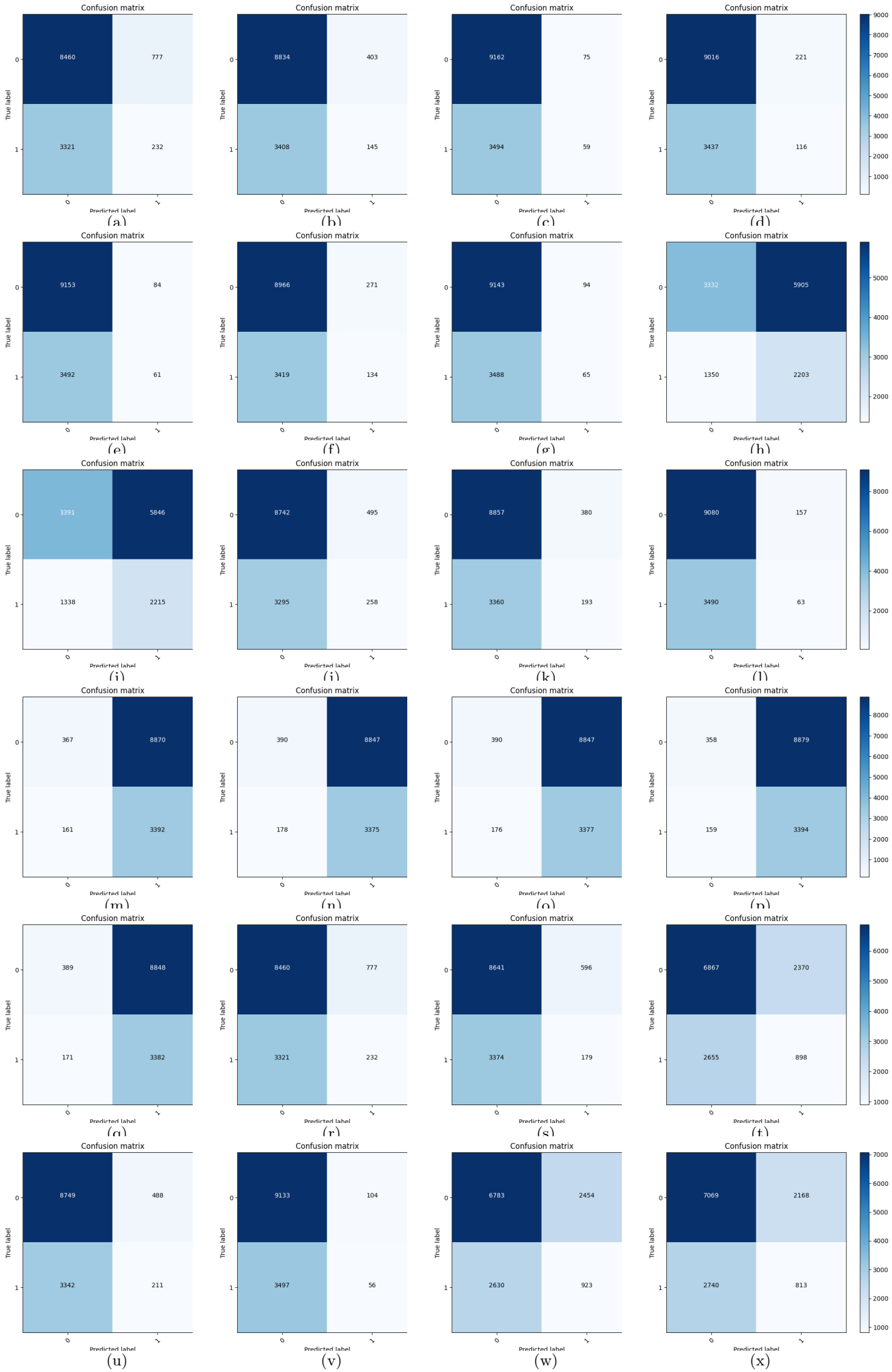


Figure 6: The models used in the above confusion matrices are respectively (a) bigram vectoriser with logistic regression; (b)