

CISC 372

Text Analytic III

Language Model & Transformer



Last lectures

- Text Analysis
 - Vanilla RNN
 - Gated Recurrent Unit
 - LSTM
 - Attention Mechanism
 - Memory
 - things we memorized
 - Context
 - based on the context, on which part of memory should we focus
 - Multi-head attention
 - Attention – Explainability
 - Padding (sorting the sequence)
 - RNN for different sequence problems

Today

- Language Model
- Neural Network for Language Model
- RNN for Language Modeling
- Transformer

Language Modelling

- Probabilities associated with language

all of a sudden I notice three guys standing on the sidewalk
on guys all I of notice sidewalk three a sudden standing the

- Predicting the next word

Please turn your homework...

Language Modelling

- Probability distribution over text words
 - Assign probability to possible next words
 - State transition
- Speech recognition
- Spelling correction
- Grammatical error correction
- Machine translation

Language Modelling

- In machine translation:

他 向 记者 介绍了 主要 内容
He to reporters introduced main content

he introduced reporters to the main contents of the statement
he briefed to reporters the main contents of the statement
he briefed reporters on the main contents of the statement

N-gram model

- N-gram modeling
- Bigram:

$$P(w_t | w_{t-1})$$

- 3-gram

$$P(w_t | w_{t-1}, w_{t-2})$$

- 4-gram

$$P(w_t | w_{t-1}, w_{t-2}, w_{t-3})$$

- N-gram

$$P(w_t | w_{t-1}, \dots, w_{t-(n-1)})$$

N-gram model

1-gram: To him swallowed confess hear both. Which.
Of save on trail for are ay device and rote life have

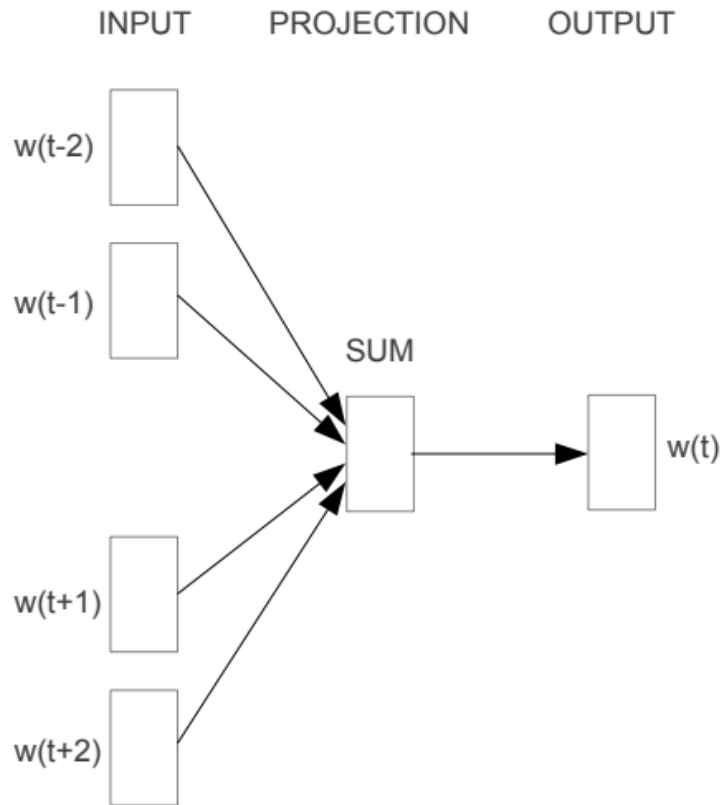
2-gram: What means, sir. I confess she? then all
sorts, he is trim, captain

- 4-gram: King Henry. What! I will go seek the traitor
Gloucester. Exeunt some of the watch. A great
banquet serv'd in
- The *longer* the context, the more coherent

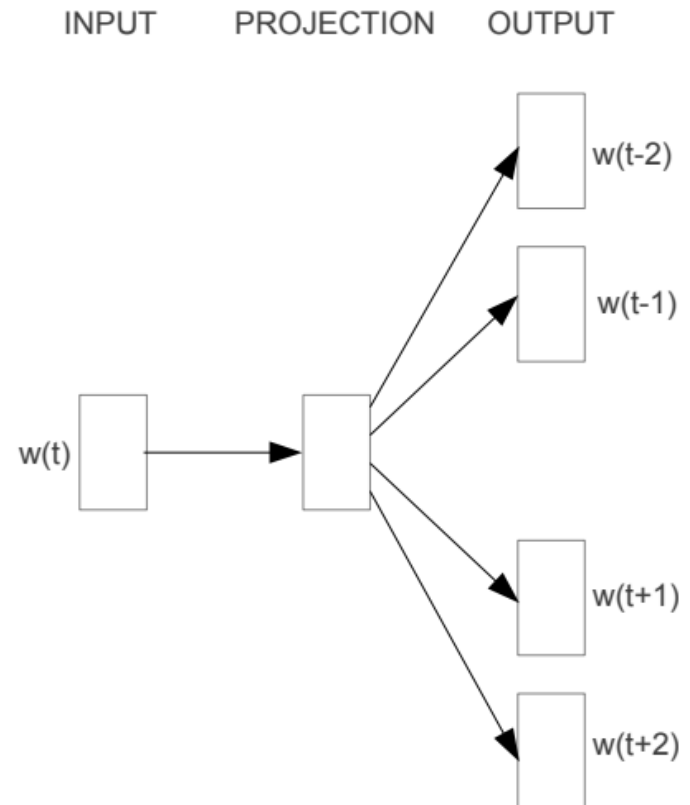
N-gram model

- Issues:
 - Storage limitation
 - Shakespeare
 - 844,000,000 bigram
 - 7×10^{17} 4-grams...
- Characteristics:
 - Sparseness

Neural Language Model



CBOW

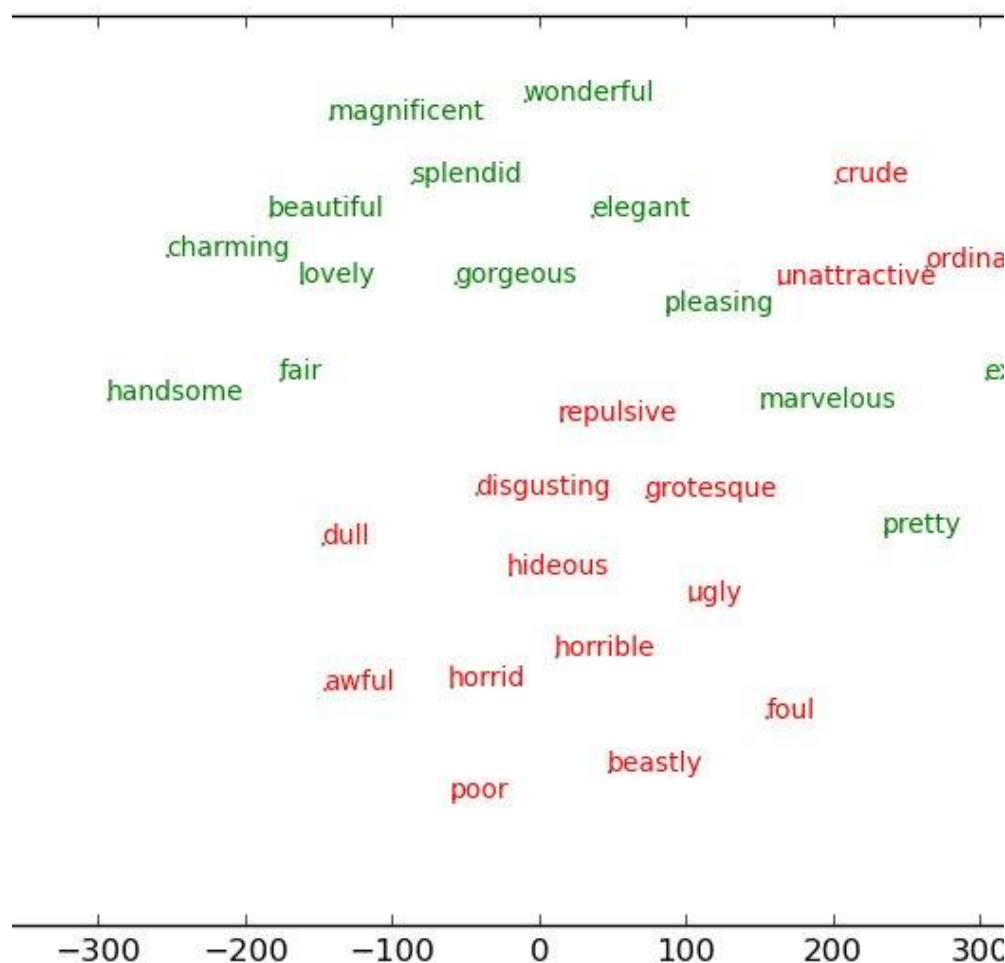


Skip-gram

Example: steven is drunk again

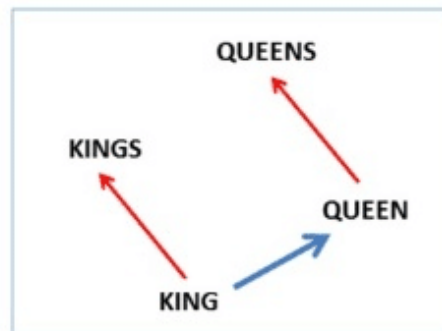
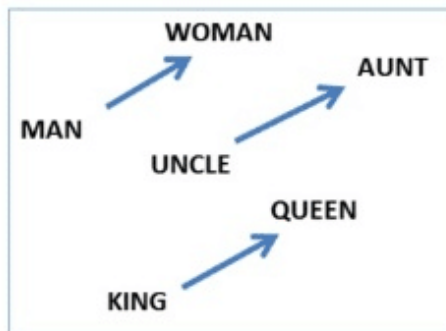
Neural Language Model

- Embedding matrix visualization (approximated nearest neighbor clustering)



Neural Language Model (vector composition)

- Learned embedding matrix is able to capture relationship between words
- Examples:
- $\text{vec}(\text{king}) - \text{vec}(\text{man}) + \text{vec}(\text{woman}) = \text{vec}(\text{queen})$



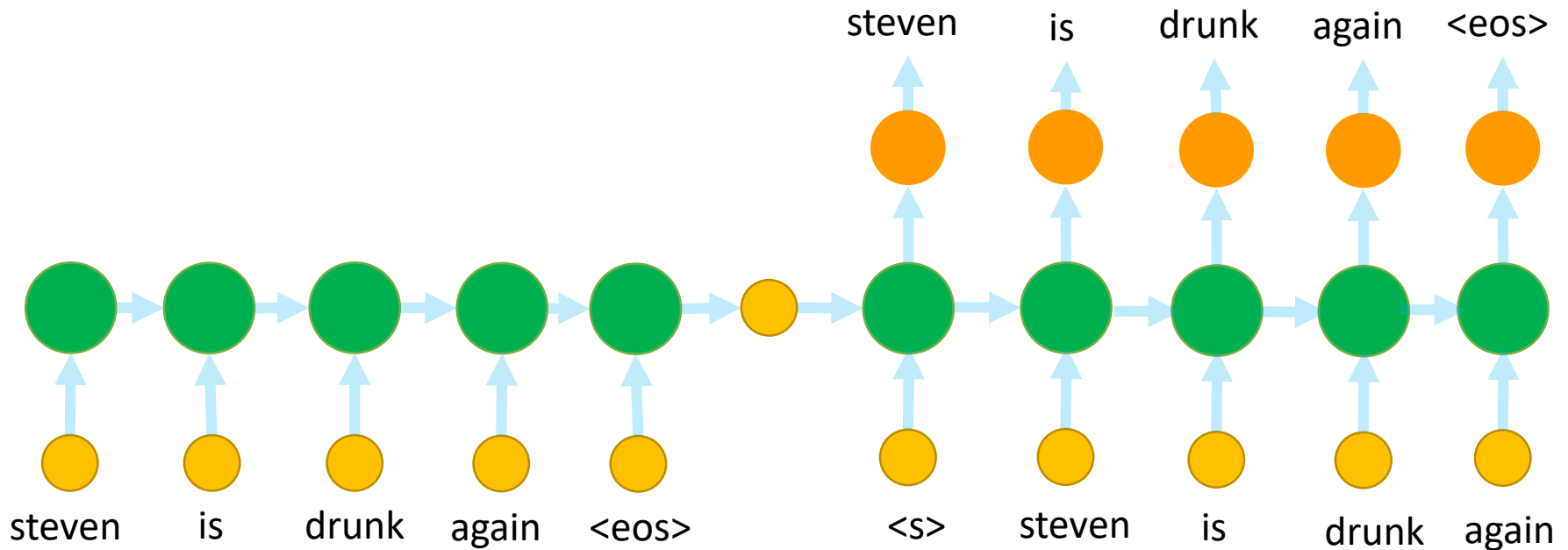
Czech + currency	Vietnam + capital	German + airlines	Russian + river	French + actress
koruna	Hanoi	airline Lufthansa	Moscow	Juliette Binoche
Check crown	Ho Chi Minh City	carrier Lufthansa	Volga River	Vanessa Paradis
Polish zolty	Viet Nam	flag carrier Lufthansa	upriver	Charlotte Gainsbourg
CTK	Vietnamese	Lufthansa	Russia	Cecile De

Neural Language Model

- Recurrent Neural Language Model
- Many-to-Many structure
- AKA encoder-decoder structure

RNN Encoder – Decoder

Example: steven is drunk again



Encoder – Decoder

- What does it learn?? (on natural language)
- Early training epochs:
 - we counter. He stutn co des. His stanted out one ofler that concossions and was to gearang reay Jotrets and with fre colt otf paitt thin wall. Which das stimn
- Later:
 - "Why do what that day," replied Natasha, and wishing to himself the fact the princess, Princess Mary was easier, fed in had oftended him. Pierre asking his soul came to the packs and drove up his father-in-law women.

Encoder – Decoder

- What does it learn?? (on source code)

```
static int indicate_policy(void)
{
    int error;
    if (fd == MARN_EPT) {
        /*
         * The kernel blank will coeld it to userspace.
         */
        if (ss->segment < mem_total)
            unblock_graph_and_set_blocked();
        else
            ret = 1;
        goto bail;
    }
    segaddr = in_SB(in.addr);
    selector = seg / 16;
    setup_works = true;
    for (i = 0; i < blocks; i++) {
        seq = buf[i++];
        bpf = bd->bd.next + i * search;
        if (fd) {
            current = blocked;
        }
    }
    rw->name = "Getjbbregs";
    bprm_self_clearl(&iv->version);
    regs->new = blocks[(BPF_STATS << info->historidac)] | PFMR_CLOBATHINC_SECONDS << 12;
    return segtable;
}
```


Encoder – Decoder

- What does it learn?? (on source code)

```
static int indicate_policy(void)
{
    int error;
    if (fd == MARN_EPT) {
        /*
         * The kernel blank will coeld it to userspace.
         */
        if (ss->segment < mem_total)
            unblock_graph_and_set_blocked();
        else
            ret = 1;
        goto bail;
    }
    segaddr = in_SB(in.addr);
    selector = seg / 16;
    setup_works = true;
    for (i = 0; i < blocks; i++) {
        seq = buf[i++];
        bpf = bd->bd.next + i * search;
        if (fd) {
            current = blocked;
        }
    }
    rw->name = "Getjbbregs";
    bprm_self_clearl(&iv->version);
    regs->new = blocks[(BPF_STATS << info->historidac)] | PFMR_CLOBATHINC_SECONDS << 12;
    return segtable;
}
```

Encoder – Decoder

- What does it learn?? (on source code)

Cell that robustly activates inside if statements:

```
static int __dequeue_signal(struct sigpending *pending, sigset_t *mask,
                           siginfo_t *info)
{
    int sig = next_signal(pending, mask);
    if (sig) {
        if (current->notifier) {
            if (sigismember(current->notifier_mask, sig)) {
                if (!(current->notifier)(current->notifier_data)) {
                    clear_thread_flag(TIF_SIGPENDING);
                    return 0;
                }
            }
        }
        collect_signal(sig, pending, info);
    }
    return sig;
}
```

Cell that is sensitive to the depth of an expression:

```
#ifdef CONFIG_AUDITSYSCALL
static inline int audit_match_class_bits(int class, u32 *mask)
{
    int i;
    if (classes[class]) {
        for (i = 0; i < AUDIT_BITMASK_SIZE; i++)
            if (mask[i] & classes[class][i])
                return 0;
    }
    return 1;
}
```

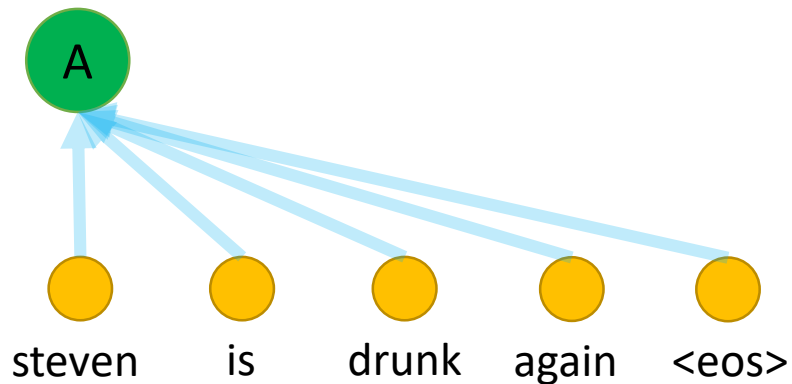
Issue

- Long-term dependencies
 - Different cell implementation
 - Attention
- Computational complexity
 - The time stamp t 's calculation
 - depends on time stamp $t-1$
 - For encoder/decoder:
 - Required *2 times t* passes over the memory cell to train a batch/sample

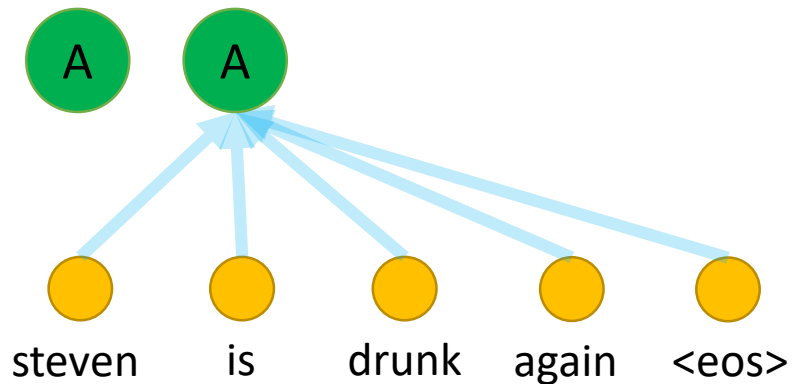
The Transformer – Attention is all you need

    
steven is drunk again <eos>

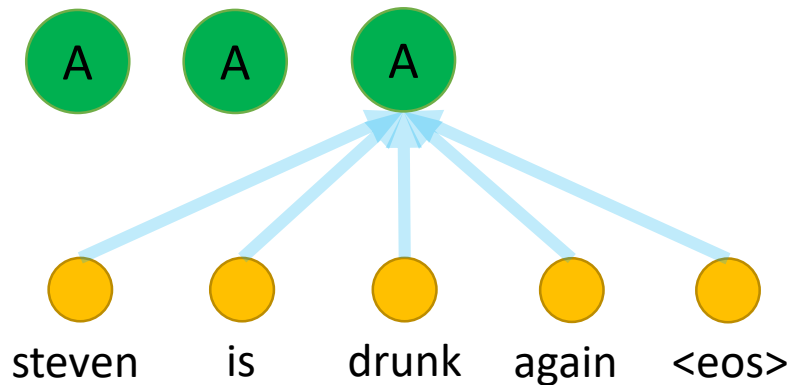
The Transformer – Attention is all you need



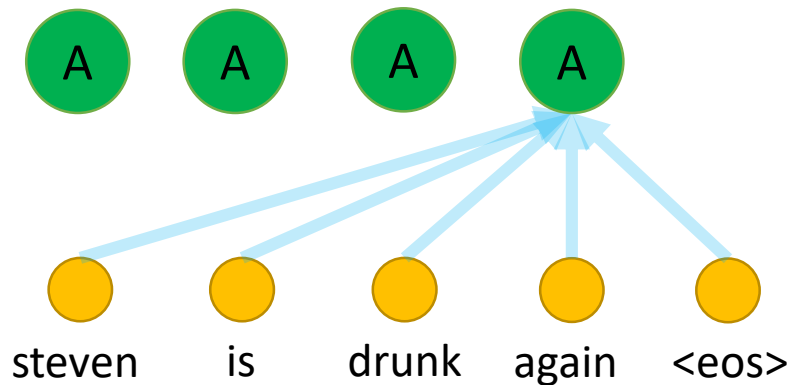
The Transformer – Attention is all you need



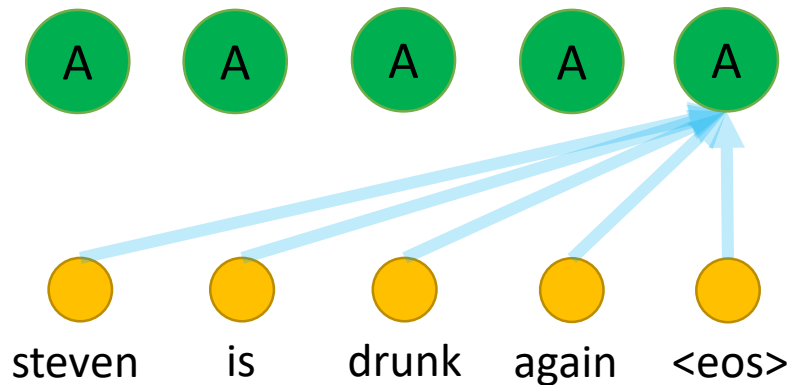
The Transformer – Attention is all you need



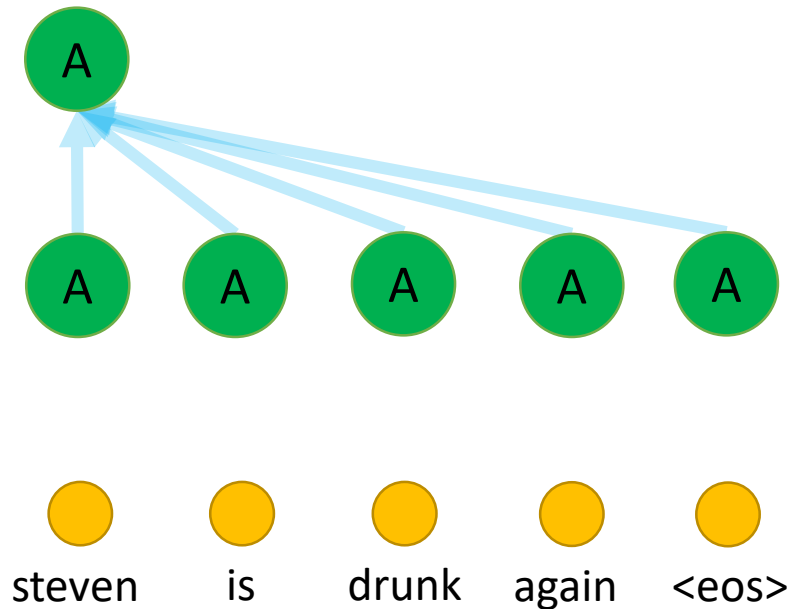
The Transformer – Attention is all you need



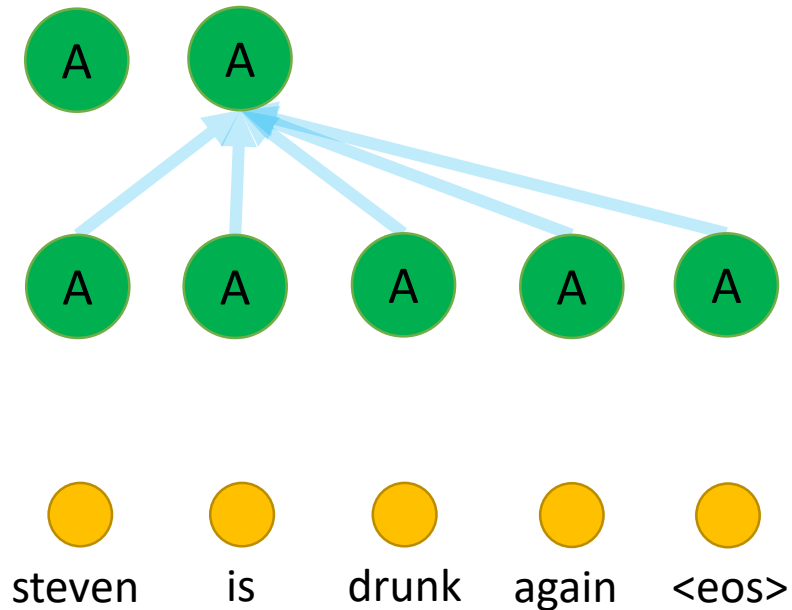
The Transformer – Attention is all you need



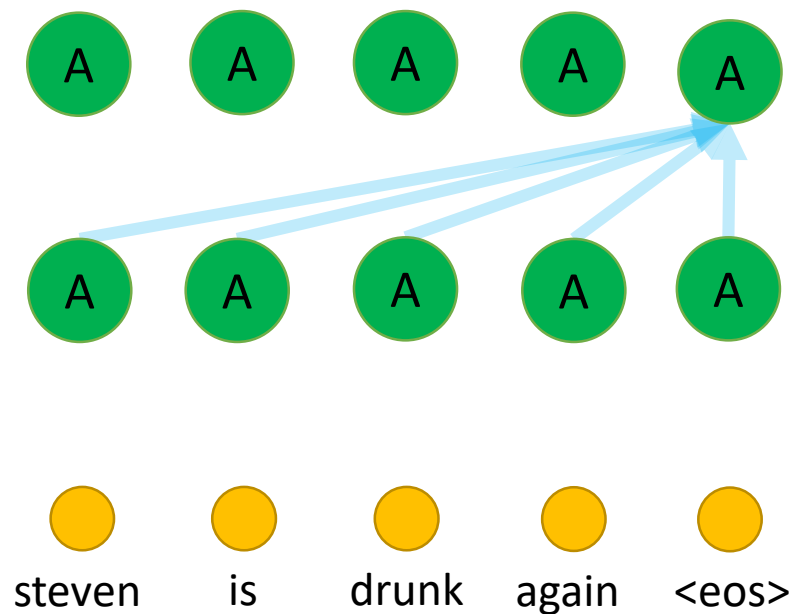
The Transformer – Attention is all you need



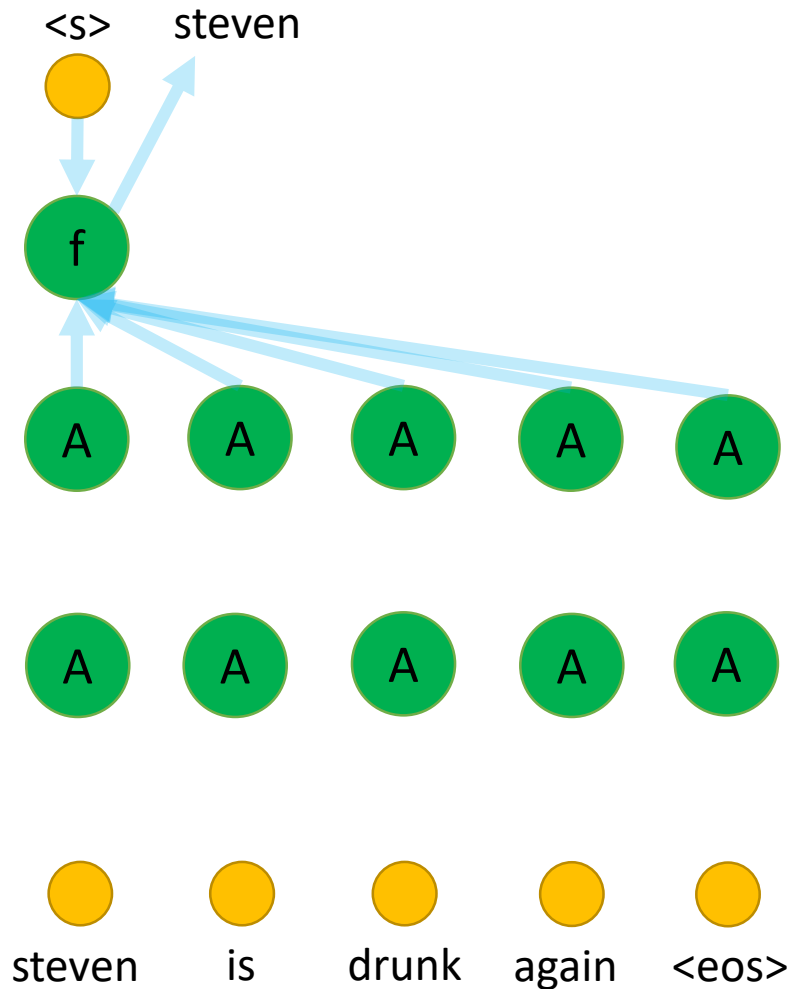
The Transformer – Attention is all you need



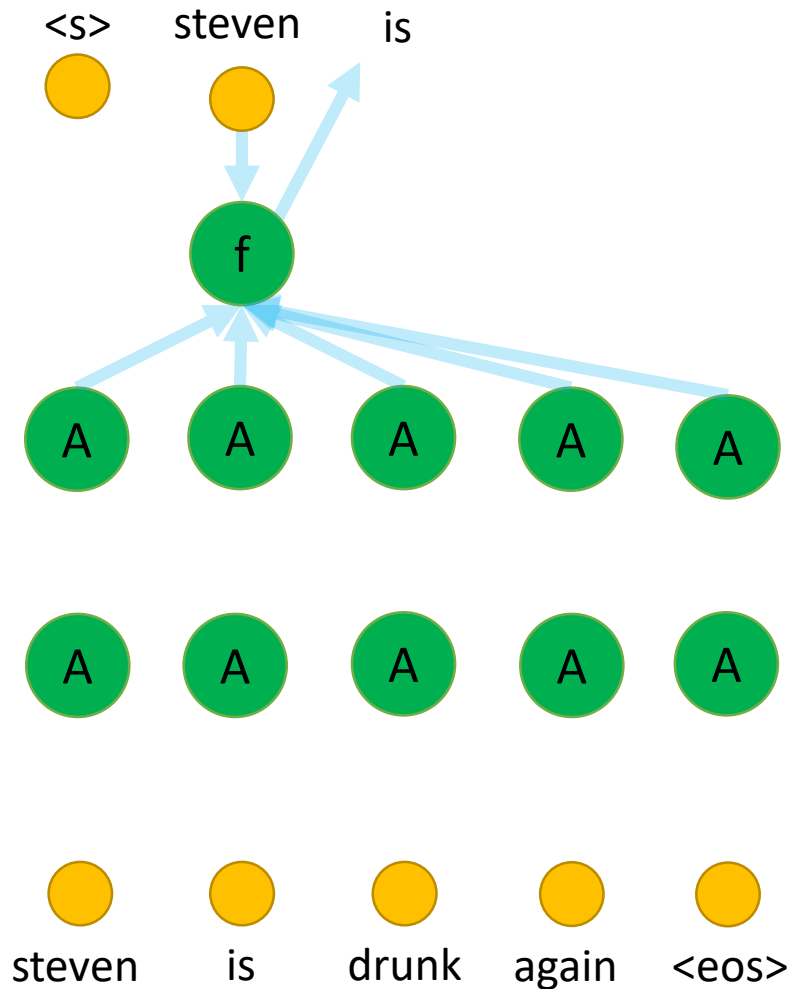
The Transformer – Attention is all you need



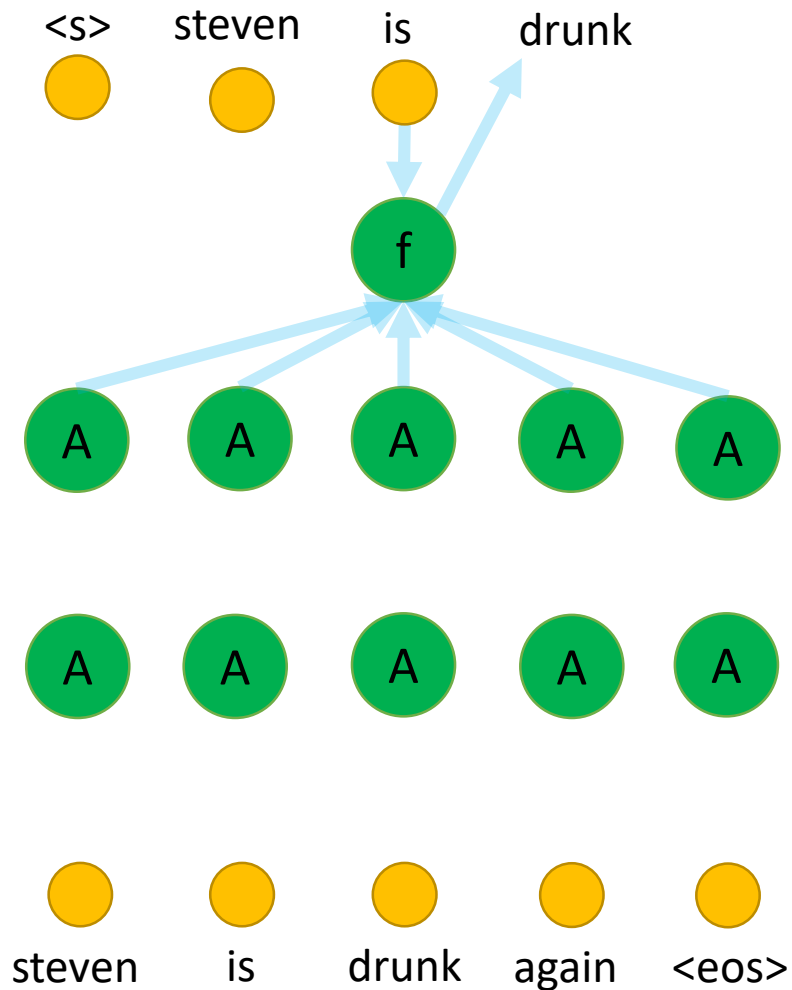
The Transformer – Attention is all you need



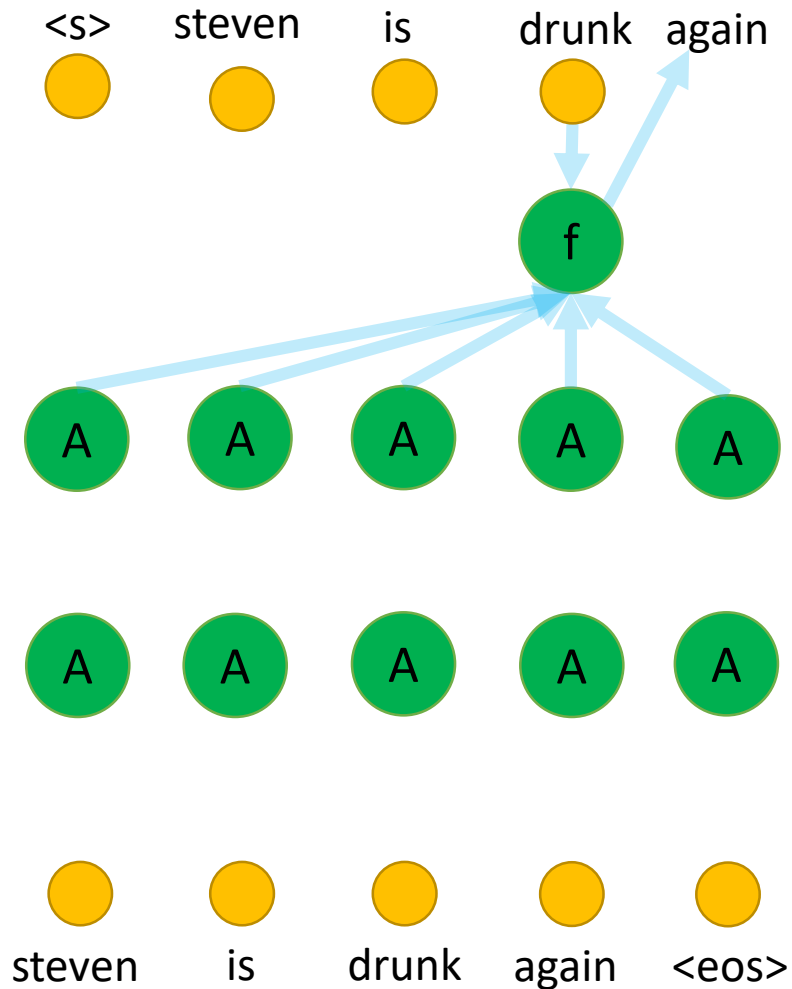
The Transformer – Attention is all you need



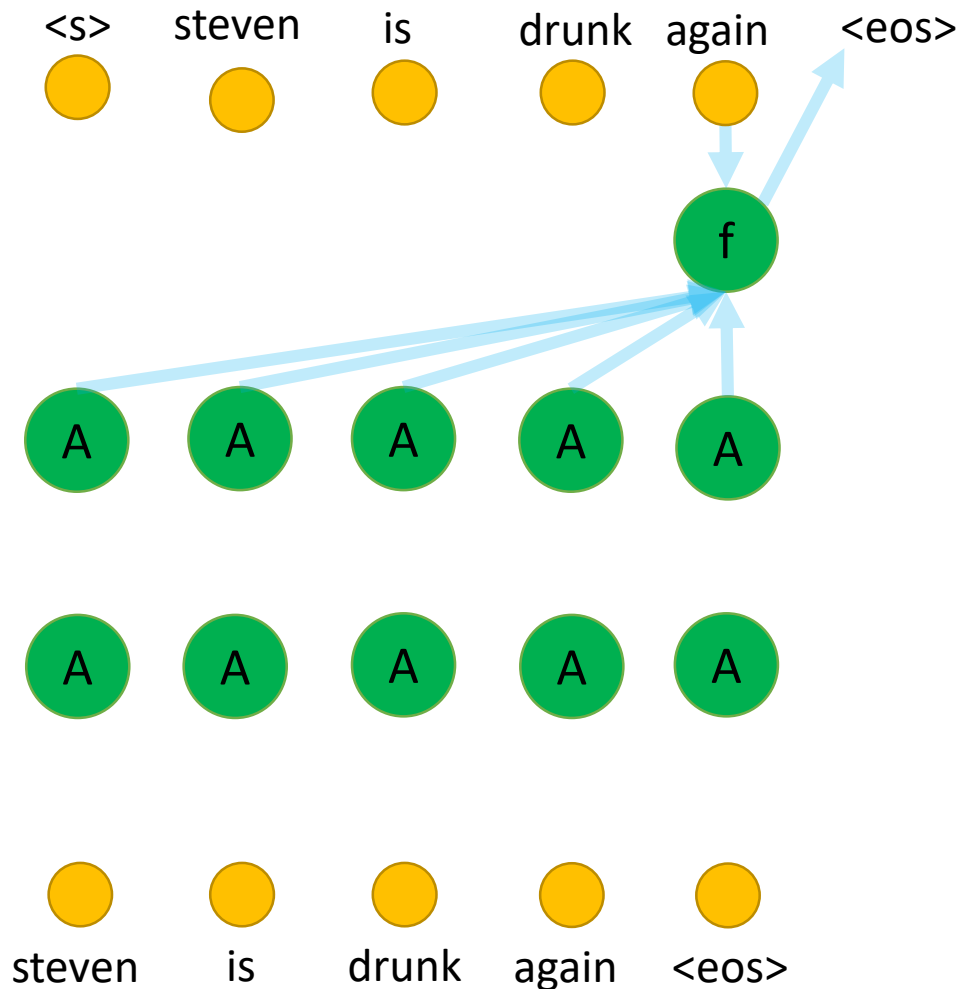
The Transformer – Attention is all you need








The Transformer – Attention is all you need



The Transformer – Attention is all you need



Performance on down-stream tasks:

	Rank	Name	Model	URL	Score	CoLA	SST-2	MRPC	STS-B	QQP	MNLI-m	MNLI-mm	QNLI	RTE	WNLI	AX
	1	T5 Team - Google	T5		90.3	71.6	97.5	92.8/90.4	93.1/92.8	75.1/90.6	92.2	91.9	96.9	92.8	94.5	53.1
	2	ERNIE Team - Baidu	ERNIE		90.1	72.8	97.5	93.2/91.0	92.9/92.5	75.2/90.8	91.2	90.8	96.1	90.9	94.5	49.4
	3	Microsoft D365 AI & MSR AI & GATECHMT-DNN-SMART			89.9	69.5	97.5	93.7/91.6	92.9/92.5	73.9/90.2	91.0	90.8	99.2	89.7	94.5	50.2
+	4	Alibaba DAMO NLP	ALICE v2 large ensemble (Alibaba DAMO NLP)		89.7	73.2	97.1	93.9/91.9	93.0/92.5	74.8/91.0	90.8	90.6	95.9	87.4	94.5	48.7
+	5	Microsoft D365 AI & UMD	FreeLB-RoBERTa (ensemble)		88.4	68.0	96.8	93.1/90.8	92.3/92.1	74.8/90.3	91.1	90.7	95.6	88.7	89.0	50.1
	6	Junjie Yang	HIRE-RoBERTa		88.3	68.6	97.1	93.0/90.7	92.4/92.0	74.3/90.2	90.7	90.4	95.5	87.9	89.0	49.3
	7	Facebook AI	RoBERTa		88.1	67.8	96.7	92.3/89.8	92.2/91.9	74.3/90.2	90.8	90.2	95.4	88.2	89.0	48.7
+	8	Microsoft D365 AI & MSR AI	MT-DNN-ensemble		87.6	68.4	96.5	92.7/90.3	91.1/90.7	73.7/89.9	87.9	87.4	96.0	86.3	89.0	42.8
	9	GLUE Human Baselines	GLUE Human Baselines		87.1	66.4	97.8	86.3/80.8	92.7/92.6	59.5/80.4	92.0	92.8	91.2	93.6	95.9	-