The model also performs well on queries with partial Freebase correlates, such as "Microsoft head honcho ___", "The United States, ___'s closest ally", and "Patriots linebacker ___," although with somewhat lower average precision. The high weight features in these cases tend to provide useful hints, even though there is no direct correlate; for example, the model learns that "honchos" are people, and that they tend to be CEOs and film producers. There are also some areas where our model can be improved. First, in some cases, the edge sequence features used by the model are not expressive enough to identify the correct relation in Freebase. An example of this problem is the "linebacker" example above, where the features for *linebacker_N/N* can capture which athletes play for which teams, but not the *positions* of those athletes. Second, our model can under-perform on predicates with no close mapping to Freebase. An example where this problem occurs is the query "___ is a NASA mission." Third, there remains room to further improve the logical forms produced by the semantic parser, specifically for multi-word expressions. One problem occurs with multi-word noun modifiers, e.g., "Vice president Al Gore" is mapped to $vice(\text{AL GORE}) \wedge president(\text{AL GORE})$. Another problem is that there is no back-off with multi-word relations. For example, the predicate *head_honcho_N/N* was never seen in the training data, so it is replaced with *unknown*; however, it would be better to replace it with *honcho_N/N*, which *was* seen in the training data. Finally, although using connected entities in Freebase as additional candidates during inference is helpful, it often over- or under-generates candidates. A more tailored, per-query search process could improve performance.

7 Related work

There is an extensive literature on building semantic parsers to answer questions against a KB (**?**; **?**; **?**; **?**). Some of this work has used surface (or *ungrounded*) logical forms as an intermediate representation, similar to our work (**?**; **?**; **?**; **?**). The main difference between our work and these techniques is that they map surface logical forms to a single executable Freebase query, while we learn execution models for the surface logical forms directly, using a weighted combination of Freebase queries as part of the model. None of these prior works can assign meaning to language that is not directly representable in the KB schema. Choi, Kwiatkowski and Zettlemoyer (**?**) presented an information extraction system that performs a semantic parse of open-domain text, recognizing when a predicate cannot be mapped to Freebase. However, while they recognize when a predicate is not mappable to Freebase, they do not attempt to learn execution models for those predicates, nor can they answer questions using those predicates.

Yao and Van Durme (**?**) and Dong et al. (**?**) proposed question answering models that use similar features to those used in this work. However, they did not produce semantic parses of language, instead using methods that are non-compositional and do not permit complex queries. Finally, learning probabilistic databases in an open vocabulary semantic parser has a strong connection with KB completion. In addition to SFE (**?**), our work draws on work on embedding the entities and relations in a KB (**?**; **?**; **?**; **?**; **?**), as well as work on graph-based methods for reasoning with KBs (**?**; **?**; **?**; **?**). Our combination of embedding methods with graph-based methods in this paper is suggestive of how one could combine the two in methods for KB completion. Initial work exploring this direction has already been done by Toutanova and Chen (**?**).

8 Conclusion

Prior work in semantic parsing has either leveraged large knowledge bases to answer questions, or used distributional techniques to gain broad coverage over all of natural language. In this paper, we have shown how to gain both of these benefits by converting the queries generated by traditional semantic parsers into features which are then used in open vocabulary semantic parsing models. We presented a technique to do this conversion in a way that is scalable using graph-based feature extraction methods. Our combined model achieved relative gains of over 50% in mean average precision and mean reciprocal rank versus a purely distributional approach. We also introduced a better mapping from surface text to logical forms, and a simple method for using a KB to find candidate entities during inference. Taken together, the methods introduced in this paper improved mean average precision on our task from .163 to .370, a 127% relative improvement over prior work.

This work suggests a new direction for semantic parsing research. Existing semantic parsers map language to a single KB query, an approach that successfully leverages a KB's predicate instances, but is fundamentally limited by its schema. In contrast, our approach maps language to a *weighted combination of queries* plus a distributional component; this approach is capable of representing a much broader class of concepts while still using the KB when it is helpful. Furthermore, it is capable of using the KB even when the meaning of the language cannot be exactly represented by a KB predicate, which is a common occurrence. We believe that this kind of approach could significantly expand the applicability of semantic parsing techniques to more complex domains where the assumptions of traditional techniques are too limiting. We are actively exploring applying these techniques to science question answering (**?**), for example, where existing KBs provide only partial coverage of the questions.

Quas veritatis voluptatem voluptate nulla ut error dolores temporibus fugiat, earum eius voluptatum?Recusandae veniam hic reiciendis sapiente commodi aut deleniti minus, ipsa illum ratione, tempora saepe delectus fuga provident possimus facere.Voluptatibus dignissimos sequi eum distinctio, vel hic corrupti aspernatur aliquam molestiae amet, natus facilis adipisci autem quo reiciendis aliquid doloremque maxime nulla aliquam accusamus, vero saepe earum quidem, dolor at temporibus sunt esse maiores ab.Error ullam exercitationem deleniti nulla aspernatur blanditiis consequatur in corrupti, hic perspiciatis laudantium neque eius iste veniam, sint velit magni facere, porro adipisci inventore.Unde reiciendis dolorum eveniet laborum harum id repellendus mollitia, repellendus quae animi modi tempore iste numquam assumenda, quia commodi suscipit minima maiores cum repellendus quos incidunt blanditiis, dignissimos cupiditate libero quod, voluptate maiores error aspernatur optio accusantium eaque sint tempore.Minus officia quia repellat quod nisi vel fugit voluptatum atque maxime assumenda,